



Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Laboratorio di programmazione

Corso di laurea triennale in Informatica

Turno A (A-C) - Turno B (D-K)

Docenti: Anna Morpurgo - Andrea Trentini
Tutor: Alessandro Minoli - Nicholas Fornaroli

Dipartimento di Informatica
Università degli Studi di Milano

A.A. 2024-2025



Osservazioni sulle consegne del Lab01 - I

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

- **Numero di studenti che hanno consegnato:** bene
- **Numero e tipo di esercizi consegnati:** bene
- **Nomi dei file:** chi bene, chi deve migliorare...
- **Nomi** delle variabili e delle costanti: chi bene, chi deve migliorare...
- **Indentazione:** chi bene, chi deve migliorare...
- **Costanti** = usatele! Non usate numeri magici (32, 60, 7)
- I **calcoli** (3600, 86400, 1.8, ...): lasciateli fare al programma (cioè scriveteli sotto forma di espressioni, es.: $60 * 60 * 24$)
- Il codice deve essere **DRY** (*Don't Repeat Yourself*): se il risultato di una stessa operazione serve più volte, salvatelo in una variabile e poi fate riferimento alla variabile stessa, non duplicate il codice



Osservazioni sulle consegne del Lab01 - II

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

- 14: n. esercizi assegnati
- 517: n. file consegnati (3 file tar/zip, qualche eseguibile)
- 43: n. studenti che hanno consegnato esercizi
- 12: n. medio di consegne per studente
- 77 nomi di file (inclusi tar, zip e eseguibili)
- 5,28 nomi in media per esercizio!
 - ❶ conversione-giorni.go
 - ❷ conversione_giorni
 - ❸ **conversione_giorni.go**
 - ❹ conversione_giorno.go
 - ❺ conversioneanni.go
 - ❻ conversioni.go
 - ❼ conversioni_giorni.go



Osservazioni sulle consegne del Lab01 - III

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Uso e non uso delle costanti

```
///// conversione secondi //////////  
const secondiMinuto = 60 //nomi maiuscoli, es.: SECONDI_MINUTO  
const secondiOra = 3600 //far fare i calcoli al computer  
const secondiGiorno = 86400 //far fare i calcoli al computer  
gg := secTot/secondiGiorno  
secondiRimanenti := secTot%86400 //usare la costante  
  
////////// centigradi - fahrenheit //////////  
const A, B = 5.0, 9.0  
gradiFahreheit := ((gradiCentigradi*B)/A) + 32 // e 32? usare le costanti  
  
///// conversione giorni //////////  
  
const anno = 365 //nomi maiuscoli, es.: GIORNI_ANNO  
const settimana = 7 //idem  
  
////////// angolo triangolo //////////  
const ang3 = 180 //usare nomi maiuscoli e pertinenti, es.: SOMMA_ANGOLI_TRIANGOLO
```



Osservazioni sulle consegne del Lab01 - IV

Laboratorio di
programmazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Aderenza alle specifiche

```
///// intersezione rette /////
/*
Esempio di esecuzione:
retta 1: m e q? 1    4
retta 2: m e q? 2    6
intersezione in (-2,2)
*/
fmt.Print("Dammi m e q della prima retta ") // rispetta l'esempio di esecuzione?
fmt.Scan(&m, &q)
fmt.Print("Dammi m e q della seconda retta ") // idem
fmt.Scan(&m2, &q2)

//////// operazioni //////////
/*
completare il seguente programma
...
fmt.Println(x1, "+", x2, "=", somma)
fmt.Println("differenza =", differenza)
fmt.Println("il prodotto di", x1, "e", x2, "dà", prodotto)
fmt.Println(x1, "/", x2, "=", quoziente)
*/
fmt.Println("Somma dei numeri: ", c)
fmt.Println("Sottrazione dei numeri: ", d)
fmt.Println("Moltiplicazione dei numeri: ", e)
fmt.Println("Divisione dei numeri: ", f)
```



Osservazioni sulle consegne del Lab01 - V

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

Calcoli

```
///// conversione secondi //////////  
gg = s / 60 / 60 / 24  
hh = s / 60 / 60 % 24 // salvare i risultati, non ripetere i conti  
min = s / 60 % 60  
sec = s % 60  
  
////////// distanza punti //////////  
math.Pow((x2-x1), 2) // per il quadrato, usare x*x: (x2-x1)*(x2-x1)
```



Uso di funzioni di libreria - I

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Premessa lessicale:

- Un'**istruzione** denota un'**azione** (ad esempio: leggi, stampa, memorizza, ecc.) ed è generalmente costituita da una riga di codice (si possono mettere più istruzioni su una stessa riga, separate da ';') .
- Un'**espressione** è una combinazione di *variabili*, *valori* e *operatori* che denota un **calcolo** e quindi fornisce un **valore**, **da memorizzare** in una variabile, **da passare** a una funzione (ad esempio di una funzione per la stampa, ma non solo) o da usare a sua volta in un'altra espressione. Un'espressione si troverà quindi all'interno di una istruzione a destra del simbolo per l'**assegnamento** (=) o come **parametro** di una funzione.



Uso di funzioni di libreria - II

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Una funzione è definita da

- nome
- numero e tipo di argomenti (anche 0)
- numero e tipo di valori restituiti (anche 0)

oltre ovviamente da quello che fa



Uso di funzioni di libreria - III

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Ci sono:

- **funzioni** che vanno usate come **espressioni**, quindi invocate ad esempio a destra del simbolo di assegnamento,
- **funzioni** che svolgono **azioni**, quindi che vanno invocate come istruzioni (righe di codice),
- **funzioni** che possono svolgere **tutti e due** i ruoli (ad esempio la funzione `fmt.Scan`).

Esempio:

```
var num int
fmt.Scan(&num) //istruzione
n, err := fmt.Scan(&num) //espressione a destra di :=
```



Il costrutto if

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

Sintassi dell'if:

```
if <condizione>{  
    blocco di istruzioni  
} else if <altra condizione>{  
    altro blocco di istruzioni  
} else if ...  
} else {  
    un altro blocco ancora  
}
```

dove <condizione> è un'**espressione booleana** (di tipo bool),
il cui valore è cioè true o false, e le **condizioni** sono
mutuamente esclusive.



Visibilità (*scope*) delle variabili

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

```
//la variabile v è locale al costrutto if
if v := math.Pow(x, n); v < lim {
    fmt.Println(v)
} else {
    fmt.Println(v, " >= ", lim)
}
```

```
if t > 0 {
    // la variabile s è locale a questo blocco
    s := math.Sqrt(2*s/a)
    fmt.Println(s) //fuori da questo blocco darebbe errore
}
```



Espressioni booleane I

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Espressione booleana: il risultato della valutazione è true o false. Servono per esprimere **condizioni** nei costrutti di controllo.

Operatori:

Operatori di confronto: $==$, $!=$, $>$, $>=$, $<$, $<=$

Operatori booleani: AND ($\&\&$), OR ($\|\|$), NOT ($!$)



Espressioni booleane II

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

Esempi di variabili e assegnamenti di tipo bool:

```
var isAdmitted = true
```

```
isPos := n > 0
```

```
isFreeOrCheap := cost == 0 || cost == 2
```

Esempi di condizioni:

esempio	in Go
variabile booleana	isAdmitted
confronto	$n > 100$
valore interno a un intervallo	$(0 < n \ \&\& \ n < 10)$
valore esterno a un intervallo	$(n < 0 \ \ n > 10)$
ha date caratteristiche	<code>local && hasLowIncome</code>



Schemi per la selezione

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

selezione a una via (if senza else)

Es.: se è buio, accendo la luce

```
if <condition> {  
    <action>  
}
```

Selezione a due vie (if - else)

Es.: se è venerdì, pesce; altrimenti carne

```
if <condition> {  
    <action one>  
} else { // condition is false  
    <another action>  
}
```



Schemi per la selezione

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

Selezione a più vie (if - else if - ... - else)

Es.: $\text{delta} > 0$; $\text{delta} < 0$; $\text{delta} = 0$, azioni diverse per ogni caso

```
if <condition1> {  
    <action 1>  
} else if <condition2> {  
    <action 2>  
}  
  
...  
} else {  
    <action n>  
}
```



Schemi per la selezione

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

Scelta sequenziale (if - else if - ... - else)

se $n > 1000$: azione 1

se $100 < n \leq 1000$: azione 2

se $10 < n \leq 100$: azione 3

se $n < 10$: azione 4

Si traduce in:

```
if n > 1000 {  
    <azione 1>  
} else if n > 100 { //e non n > 100 && n < 1000  
    <azione 2>  
}  
...  
} else {  
    <azione n>  
}
```




Schemi per la selezione

Laboratorio di
programmazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

Azioni indipendenti (if - if - ... - if)

Es.: se freddo -> accendo stufa; se sabato -> invito amici; se frigo vuoto -> spesa

```
if <condition 1> {  
    <action 1>  
}  
if <condition 2> {  
    <action 2>  
}  
...  
if <condition n> {  
    <action n>  
}
```



Schemi per la selezione

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02
Rimandi
Argomenti
Schemi per la selezione

Condizioni indipendenti (if annidati)

Es.: rettangolo /cerchio; colorato/bianco;

```
if <condition 1>{  
    if <condition 2>{  
        <action 1>  
    } else {  
        <action 2>  
    } else { //! condition 1  
        if <condition 3>  
            <action 3>  
        } else {  
            <action 4>  
        }  
    }  
}
```



Pragmatica

Laboratorio di
program-
mazione - Ed. 1
- 2024/25

A. Morpurgo,
A. Trentini

Laboratorio 02

Rimandi

Argomenti

Schemi per la selezione

- Short case first
- Positive condition
- Indent for structure